REMARKS/ARGUMENTS

Pending Claims

Claims 1-9 are pending in this application. Claim 2 has been canceled without prejudice or disclaimer. Claims 1, 4-6 and 8 have been amended. No new matter has been added.

Applicants thank the Examiner for noting that this application names joint inventors and confirm the Examiner's presumption that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made.

Claim Rejections under 35 U.S.C. §103

Claims 1 and 5 have been rejected under 35 U.S.C. §103(a) as being unpatentable over Schwarm et al., U.S. Patent No. 7,028,218 in view of Hashemi, U.S. Patent No. 5,491,787. Claims 2-4 and 6-9 have been rejected under 35 U.S.C. §103(a) as being unpatentable over Schwarm et al., U.S. Patent No. 7,028,218 in view of Hashemi, U.S. Patent No. 5,491,787 as applied to claim 1 and further in view of Yasuda, U.S. Patent No. 7,069,473. Applicants request reconsideration of the rejections in view of the amendments made to claims 1, 4-6 and 8 for the following reasons.

Amendments were made to claims 1, 4-6 and 8 to more clearly describe the features of the present invention. Particularly, amendments were made to the claims to more clearly recite that the present invention is directed to a computer system including a first OS and a second OS

executed by one CPU whereby a multi-OS controller controls such that the first OS and the second OS operate independently from each other. Further, the claims have been amended to more clearly recite that the multi-OS controller of the present invention includes a memory acquisition controller that acquires the location of memory which is used by the first OS for storing the state information and the operation recording information of the first OS. Support for these amendments can be found in the specification as follows. The multi-OS controller 5 is shown in Fig.2 and page 11, lines 10 to 24 in the specification. The memory acquisition controller 8 is shown in Fig. 2 and is described in the specification from page 11, lines 25 to page 12, line 6. The operation of the memory acquisition controller is shown in page 14, lines 1 to 15 and from page 15, lines 25 to page 16, line 6. The OS state information 11 and the operation recording information 12 are shown in Fig. 3, from page 7, line 27 to page 8 to line 22 and from page 15, line 25 to page 16, line 6.

According to the present invention, as set forth in independent claim 1, for example, a computer system (and/or each computer in claim 4) includes a first OS and a second OS executed by one CPU (Fig. 1). A multi-OS controller controls such that the first OS and the second OS can operate independently from each other. For example, a business application operates on the first OS while an analysis and prediction application operates on the second OS as shown in Fig. 2. Further, according to the present invention, the memory is divided into two portions for the first OS and the second OS. In the present invention, the analysis and prediction application (operating on the second OS) can acquire the contents in the memory used by the first OS. To accomplish this function, the multi-OS controller 5 includes the

memory acquisition controller 8, which acquires the location of the memory used by the first OS for storing the OS state information 11 and the operation recording information 12 of the first OS. The auxiliary program conducts the processing against a failure of the first OS by referring to contents of the processing list on the basis of the detected failure sign.

Further, Applicants have amended independent claim 8 in the first step to clarify that the execution of the first OS and the second OS is in parallel and independent from each other. In a second step as set forth in claim 8, the contents of state information and operation recording information held by the first OS for detecting a sign of a failure, is analyzed by using an analysis and prediction application operating on the second OS. In amending claim 8, a third step has been added that sets forth that the location of memory which is used by the first OS for storing the state information and the operation recording information of the first OS is acquired. Additionally, a fourth step of processing against a failure of the first OS by referring to contents of a processing list on the basis of the detected failure sign has been added. Claim 8 further sets forth that another computer system is notified of the detected sign of the failure.

The present invention addresses software errors through use of the auxiliary driver 7 on the first OS. Unlike the references of record, the first OS and the second OS share the same hardware. Further, the method of the present invention is directed towards detecting software errors. In particular, the auxiliary driver 7 addresses software faults and errors in the first OS.

Schwarm fails to teach or suggest that the first OS holds state information and that the analysis and prediction application analyzes contents of information held by the first OS detecting a sign of failure. Schwarm discloses a two dual processor system. Two logical CPUs

on different processors are paired in a master-slave relationship as shown in Fig. 5 and as described in column 10, lines 15 to 27. The system executes a heartbeat scheme (see page 1, lines 14-22 of the present specification which describes a conventional heart beat scheme) to detect the fault of the master controller (processor). The master controller is checked from the slave controller as shown in Fig. 10 and the master controller is only monitored externally. Further, the master controller and the slave controller are different hardware.

In contrast, the present has a memory used by the first OS that is directly read by the program operating on the second OS and the both OSs are executed by the same hardware (same CPU). The first OS records its state information and its operation recording information on the memory while the analysis and prediction application operating on the second OS directly reads the information recorded by the first OS. Schwarm does not disclose these aspects of the claimed combination. Further, the deficiencies of Schwarm are not supplied by any of the other references of record.

Hashemi is relied on for disclosing a fault tolerant system in which one processor is utilized for operational purposes and the other processor would be used for checking operation of the first processor for error or fault detection. Accordingly, Hashemi is deficient in the same manner as Schwarm. That is, Hashemi discloses a multiprocessor system in which one processor is acting as a master processor and another processor is acting as the slave processor that checks the operation of the master processor. If it is detected that the master processor has failed, the slave processor becomes to the master processor in order to continue the operation. In contrast, the multi-OS controller 5 of the present invention exercises control so as to make it

possible for the first OS 1 and the second OS 2 operating on the multi-OS controller 5 to execute various kinds of processing independently of each other. Therefore, the Hashemi reference includes two separate processors while the present invention includes the two separate OSs executed by one CPU. Thus, Hashemi is different from the present invention and the reference does not make up for the deficiencies in Schwarm. Accordingly, the combination of Hashemi and Schwarm fails to teach or suggest to one having ordinary skill in the art the above described aspects of the present invention as recited in the claims. Therefore, the rejection of claims 1 and 5 under 35 U.S.C. §103(a) should be withdrawn.

Yasuda is also deficient in the same manner as Schwarm and Hashemi. In particular, Schwarm does not disclose an auxiliary program that conducts processing against a failure of the first OS. Yasuda does not make up for this deficiency.

According to the Office Action, Yasuda is relied on for teaching a storage section for storing and holding fault recovery information including a rule for defining recovery operations when a fault occurs in the computer system. However, Yasuda merely discloses a method for dealing with a hardware fault, and is not directed to dealing with software faults. Yasuda discloses that when the fault monitoring apparatus detects a hardware fault, it retrieves the previously set fault recovery rules and instructs the computer systems so as to perform a recovery operation. Yasuda discloses that the fault is notified from the fault monitoring agent in the computer system as shown in Column 13, lines 49 to 52 of the reference. Further, the fault recovery operation that is performed is mainly the change of the hardware configuration.

Therefore, the combined teachings of Schwarm, Hashemi and/or Yasuda when taken in

the manner suggested by the Examiner in the Office Action still fail to teach or suggest the claimed combination of the present invention as set forth in the pending claims. Accordingly, the rejection of claims 3-4 and 6-9 under 35 U.S.C. §103(a) as being unpatentable over Schwarm in combination with at least one of Hashemi and Yasuda should be withdrawn.

Applicants have studied the remainder of art of record and not that none overcomes the above-noted deficiencies in the references relied upon in the rejections of claims 1and 3-9.

Conclusion

In view of the foregoing, Applicant respectfully requests that a timely Notice of Allowance be issued in this case.

Respectfully submitted,

MATTINGLY, STANGER, MALUR & BRUNDIDGE, P.C.

By _____
John R. Mattingly
Reg. No. 30,293
(703) 684-1120

JRM/so
Date: January 19, 2007